(12) EUROPEAN PATENT APPLICATION

(21) Application number : 93301288.2

(51) Int. Cl.⁵ : **G06F 1/00**

(22) Date of filing : 22.02.93

(30) Priority : **17.03.92 GB 9205774**

(43) Date of publication of application :
22.09.93 Bulletin 93/38

(84) Designated Contracting States :
BE DE FR GB

(71) Applicant : **INTERNATIONAL COMPUTERS LIMITED**
**ICL House**
Putney, London, SW15 1SW (GB)

(72) Inventor : **Duxbury, Paul**
36 School Lane, Brereton
Sandbach, Cheshire CW11 9RN (GB)

(74) Representative : **Guyatt, Derek Charles**
Patents and Licensing International
Computers Limited et al
Six Hills House London Road
Stevenage, Herts, SG1 1YB (GB)

(54) Computer system security.

(57) A computer system is described in which users can access a protected resource only by way of a call to a user monitor command, specifying the protected resource as a parameter. The user monitor command checks that certain conditions are satisfied and performs specified actions before permitting access to the protected resource. The checks may include checking whether options and argument values supplied by the user satisfy specified conditions. The actions may include dynamically modifying a supplementary groups list of a current process temporarily granting or removing privileges to or from the user.

Fig.1.
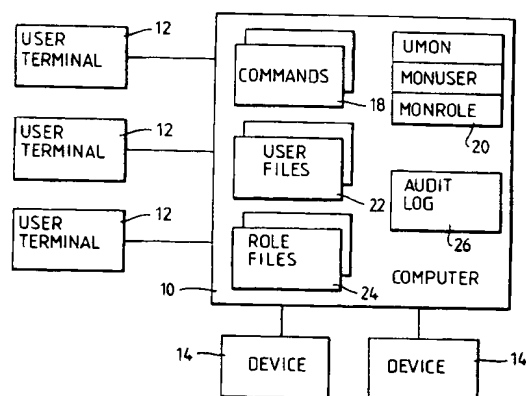
EP 0 561 509 A1

Background to the Invention

This invention relates to computer system security. More specifically, the invention is concerned with a mechanism for protecting a computer system from its users.

5   There are many ways an authorised computer user can misuse a computer system, either accidentally or maliciously. Consequences can include temporary damage to local service availability, direct or indirect financial loss, or even criminal or civil liability on the part of the employer.

Some specific threats include:

import of unauthorised software, such as games, with the inherent risk of virus attack, copyright infringement, 10 or just time wasting;

misuse of a privilege such as "superuser", which is needed to do a routine operational job, for illicit purposes like accessing private data;

disuse of password protection because of the need to know several passwords and to have some passwords shared;

15 loss of individual accountability because of the need for shared usernames and passwords;

destruction of a database, or bypass of its security, by use of the wrong programs;

The object of the present invention is to provide a mechanism which helps to protect against these threats.

Summary of the Invention

20

According to the invention there is provided a computer system accessible to a plurality of users and having a plurality of protected resources, wherein users can access a protected resource only by way of a call to a user monitor command, specifying the protected resource as a parameter, and wherein the user monitor command checks that predetermined conditions are satisfied before permitting access to the protected resource.

25

Brief Description of the Drawings

Figure 1 is a schematic block diagram of a computer system embodying the invention.

Figure 2 is a flow chart showing the operation of a user security monitor command (referred hereinafter 30 as "umon").

Description of an Embodiment of the Invention

One computer system in accordance with the invention will now be described by way of example with ref- 35 erence to the accompanying drawings.

Referring to Figure 1, the system comprises a central computer 10 which in this example is a conventional computer running under the UNIX operating system. (UNIX is a trademark of Unix Laboratories Inc).

A number of (human) users can access the central computer by way of user terminals 12. When a user wishes to access the system, the user must first log in to the system in the conventional manner. Each user 40 has a unique user identity allocated by the system.

The system also includes a number of devices 14, such as printers or data storage units, attached to the central computer.

The computer has a system memory which holds both programs and data, including a number of conventional UNIX commands 18.

45 The system also includes three special commands 20 as follows:

umon   - user security monitor

monuser   - administer umon user database

monrole   - administer umon role database

As will be described, umon makes use of a private umon database, consisting of a set of user files 22, and a 50 set of role files 24. The monuser command is used to create, update or delete the user files, while the monrole command creates, updates or deletes the role files.

The system also includes an audit log 26 which is used to record incidents in the system such as potential security violations.

The umon command enables fine control over which users can execute which commands, and with what 55 privilege. It can also control access to other resources, and can selectively log accesses.

To use umon, the command or other resource to be protected is excluded from general access by users, via the normal UNIX mechanisms (eg restrict access to owner and/or group only). The command cannot then be executed directly by a normal user. Instead, the user must call umon, passing the target command and ar-

guments as arguments to umon.

Alternatively, as well as being restricted, the target command may be moved and replaced by a symbolic link to umon. Thus any attempt to invoke the command will automatically invoke umon (which then determines the intended command from argv[0]). This invisible mode of operation allows umon to be used within suitable existing programs such as menu shells.

Umon is a privileged command controlled by the private umon database 22, 24. This database defines relevant users and the restricted command they are allowed to access. On entry, umon looks up the calling (real) user in its database to see if access is to be granted and, if so, how the temporary access is to be achieved. For example, this might be done by changing the current user and/or group, or by adding a group to the caller's list of supplementary groups.

Such access is granted for the duration of the target command only. As well a giving access to commands, temporary access to other restricted resources such as files and devices can be granted. Privileges may also be granted, eg by allowing selected commands issued by selected users to run as superuser.

In addition to restriction based on the caller's (real) login name, many other checks can be included in the database. For example: time of day; day of week; current terminal; arguments supplied by caller; and so on. Also, in addition to actions needed to grant temporary access, other actions may be defined in order to set up the environment for the command.

For ease of administration, command definitions in the database can be grouped together into named roles, representing particular user or administrative job functions. Granted permission for a user to access a role gives access to all the commands in the role, subject to other checks included with the commands. When calling umon, the role may be stated explicitly. Otherwise, the user and any roles permitted to him/her are searched in turn. A reserved role name "global" is used for commands available to everyone, and is searched last.

Umon may optionally be used without parameters, as a shell. In this case, a default command is associated with the user in the umon database.

Umon may optionally be called in a test mode. In this mode, the command will not be executed, but all checks will be performed to test whether access is available to the current user. This is useful for menu systems which require to display only those commands which are available to the current user at the current time. The umon database may also contain items which do not correspond to commands, and are included only for use with the test function, for controlling menus. For example, these may include intermediate menu names and communications path names.

Each user file 22 in the umon database relates to a particular user, and contains the following records:
- default command. This indicates the command which is to be executed in the case where umon is called without any arguments (ie as a shell).
- role list. This is a list of roles permitted for the user.
- log command list. This is a list of commands whose execution is to be recorded in the audit log if accessed by the user.
- command description. Each user file may contain a number of command descriptions, one for each target command that the user is permitted to access via umon. Each command description specifies the conditions for permitting access to the command, and also specifies how this access is to be achieved.

Each command description record has the following structure:

Cmdname[#tag][[? | !]condition[[? | !]...][:path][:action[:...]]

Cmdname is the name of the target command as it will be invoked through umon. It may be the name of the target command file, or some other suitable key.

#tag is an optional string of characters used to distinguish this record from others for the same cmdname and user (ie if the same command name is included more than once with different conditions). The tag is only needed for removing command records.

?Condition and !Condition are optional extra checks which must be performed before allowing the user access. Several conditions may be included, each beginning with either ? if the expression is to evaluate to TRUE, of ! if the condition is to evaluate to FALSE, for access to be granted. All conditions must succeed for access to be granted (ie there is a logical AND between entries). For example, the following conditions may be specified:
- check that the current terminal name is one of a specified list of values.
- check that the current real user identity matches one of a specified list of values, using a specified relational operator (eg equals, greater than, less than, not equals).
- check that the current time of day or day of the week matches one of a specified list of values.
- check that any options or argument values supplied by the user match one of a specified list of values.
:Path is the full path name for the target command.

3

If the path name is omitted, the record may only be used with the -t (test) option of umon (eg to record a menu name or a service name). If no path field is included, the command description may be accessed via the test mode of umon. Additional conditions are allowed and are evaluated during test, but no additional actions may be included. Thus, menu names and service names can be restricted in the same way as commands, though this must be enforced by the menu system itself (umon here performing an advisory function).

:Action is an optional action to perform before executing the target command. Several actions may be included, and are executed in sequence. Usually, actions will be used to gain access to the target command, or to acquire other privilege such as superuser. For example, the following actions may be specified:

- set the real and effective user or group of the current process to a specified·value, eg superuser.
- replace the user-specified argument list with a specified argument list.
- change the current directory or root to a specified path.
- dynamically add a specified list of groups to the current processes supplementary groups list, to give temporary access to new resources, such as a device or the target command itself, without removing any existing access permissions.
- remove one or more groups from the current process's supplementary groups list. This can be used to remove access/privilege, for example when entering a shell, to prevent the user breaking out to perform other commands with this same privilege.
- set the user ID back to that of the original user at session login, eg to remove superuser status. Again, this allows privilege to be reset when entering command shells.
- establish a read or write lock on a specified file path.
- execute a specified command or shell script.

Each role file 24 in the umon database relates to a particular role which has been defined in the system, and contains the following records:

- log command list.
- one or more command descriptions.

These are similar to the corresponding records in the user files.

The form of a call to the umon command is:

umon [-t] [role.] cmd [arg...]

where square brackets indicate optional parameters.

The presence of the -t parameter indicates that the command is to run in test mode. The role parameter allows a user to specify a role name explicitly. This is useful in the case where a user has access to the same command by way of more than one role but with different permissions and/or actions. Normally, the first occurring role would be used, which might not reflect all the privileges that the user requires. The other parameters specify the target command, and the arguments of the target command.

The operation of the umon command will now be described in more detail with reference to Figure 2.

(2-1) When call is made to umon, umon first consults the umon database to find the user file (if any) for the current user.

(2-2) A check is then made to determine whether a role was explicitly specified.

(2-3) If no role was specified, umon searches the user file to find a command description record that relates to the specified target command. Umon then checks whether the conditions specified in this record are satisfied. If so, then access to the target command by the user is permitted. If, on the other hand, no command description record for the target command is found, or the conditions are not satisfied, access is denied.

(2-4) If access is permitted, umon then checks whether the -t option is specified in the call. If the -t option was specified, umon simply returns a "0" value, without performing any further actions.

(2-5) If the -t option was not specified, umon then performs any actions specified in the actions field of the command description, to set up the environment for execution of the target command.

(2-6) The target command is then executed. The target command execution is direct (using "exec") rather than via shell (using "system") to avoid the possibility of "Trojan horse" commands being supplied as arguments to the original command.

Umon then returns the result of the execution of the target instruction to the user.

(2-7) If a role was explicitly specified, umon checks that this role is on the user's role list. If so, umon accesses the role file for this role, and determines whether this role is permitted to access the target command. If access is permitted, umon continues at step (2-4) as described above. If the role is not on the user's role list, or the role is not permitted to access the target command, umon returns a value -1 to indicate failure.

(2-8) If user access was not permitted at step (2-3), umon then consults the role list in the user file, to determine which roles are permitted for this user. Umon then consults the role file for each of these per-

4

mitted roles in turn, and checks whether that role is permitted to access the target command. If a role is found for which access is permitted, umon continues at step (2-4) as described above.

(2-9) If no user file was found at step (2-1), or no suitable role file was found at step (2-8), umon then accesses the global role file and checks whether the global role is permitted to access the target command. If so, then umon continues at step (2-4) as described above. If the global role is not permitted to access the target command, umon returns a value -1 to indicate failure.

In summary, it can be seen that umon helps protect against threats by dynamically controlling the commands, command arguments, privileges, devices and files available to any user. Thus, a user can be restricted to a defined set of commands, and any privileges needed to execute a command can be allocated dynamically for the duration of the command only.

For example, a user who needs to perform filestore backup can be granted access to the prescribed backup program from his own login. He can be granted superuser privilege and access to the exchangeable media device only for the duration of that program. He can be denied access to other privileged commands, such as restore. This avoids him using superuser privilege to access confidential information, or using the exchangeable media device to import illicit software, or needing to know more than one password.

Umon is role-based for convenience, supports command-line and menu-based access, can be used invisibly from within existing programs, and allows selected actions to be logged and traced back to a specific end-user by means of the audit log.

Claims

1. A computer system accessible to a plurality of users and having a plurality of protected resources, characterised in that users can access a protected resource only by way of a call to a user monitor command, specifying the protected resource as a parameter, and wherein the user monitor command checks that predetermined conditions are satisfied before permitting access to the protected resource.

2. A system according to Claim 1 wherein the checks performed by the user monitor command include checking whether options and argument values supplied by the user satisfy specified conditions.

3. A system according to Claim 1 or 2 wherein the user monitor command performs predetermined actions before access is permitted in order to ensure that access is made in a controlled manner.

4. A system according to Claim 3 wherein said actions include dynamically modifying a current process's supplementary groups list.

5. A system according to Claim 3 or 4 wherein said actions include temporarily granting or removing privileges to or from the user.

6. A system according to any preceding claim wherein the user monitor command makes use of a private database indicating which users are permitted to indirectly access which resources.

7. A system according to Claim 6 wherein the database also indicates the conditions that are to be satisfied and actions that are to be performed before permitting access.

8. A system according to Claim 6 or 7 wherein the database also indicates which of a predetermined set of roles are available to each user, and indicates which roles are permitted to access which resources.

9. A system according to any preceding claim wherein the resources comprise commands.

10. A system according to any preceding claims wherein said call to the user monitor command may be made indirectly by way of a symbolic link.

5

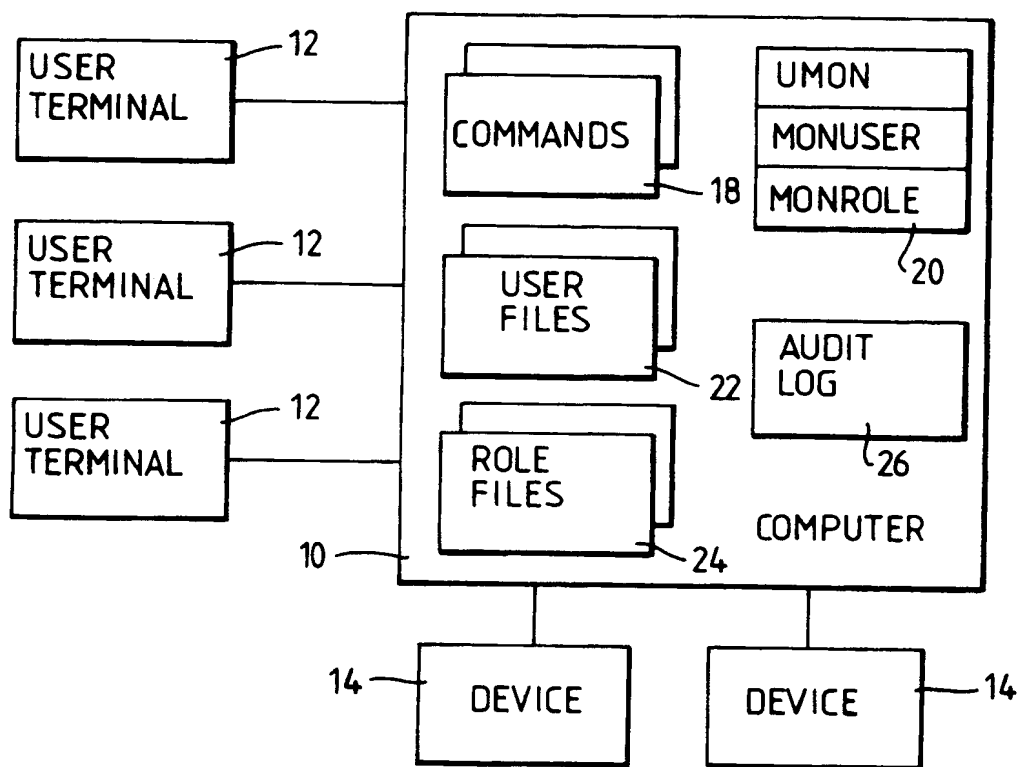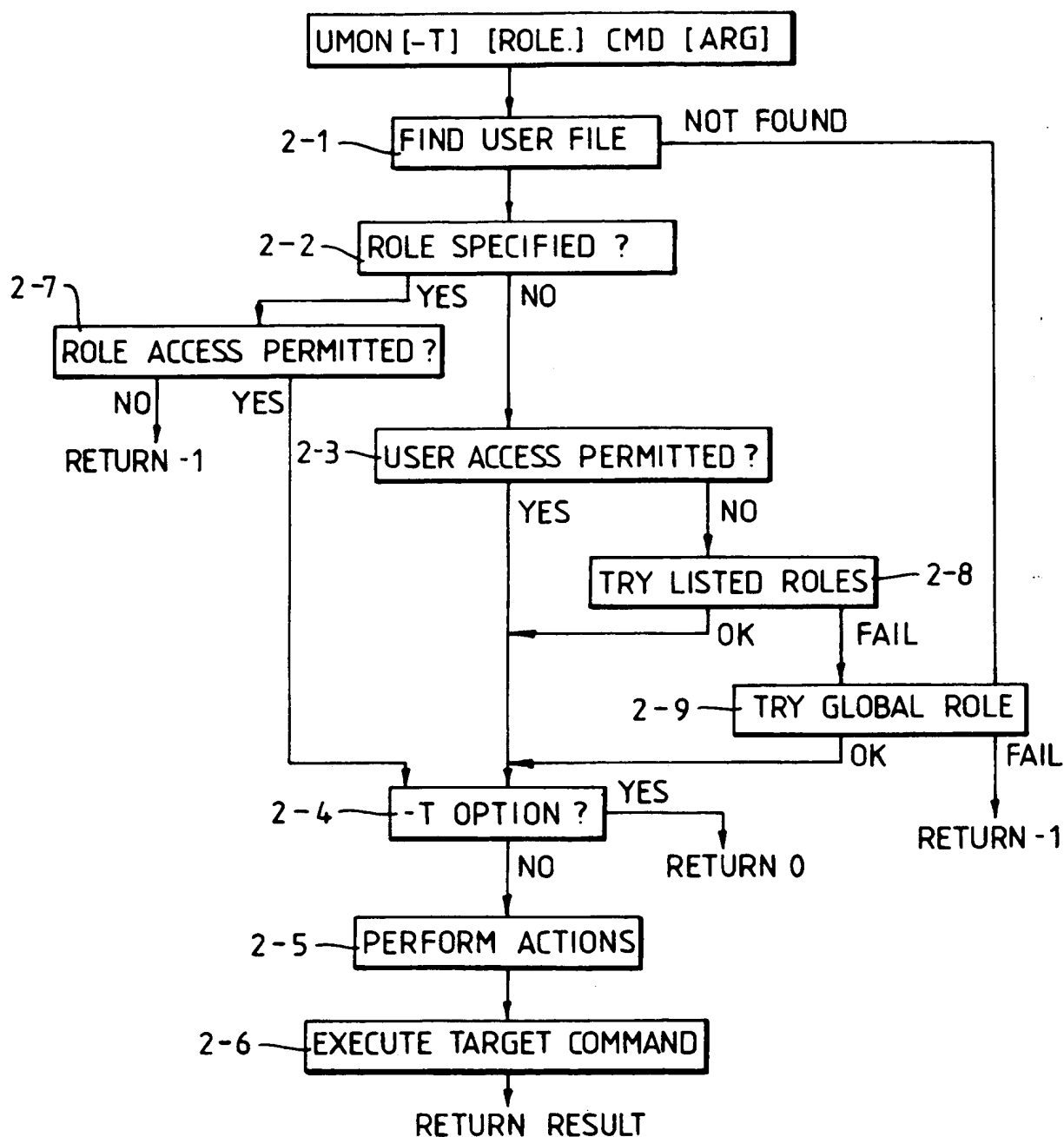# Fig. 1.

Fig.2.

UMON [-T] [ROLE.] CMD [ARG]

2-1 — FIND USER FILE — NOT FOUND

2-2 — ROLE SPECIFIED ?
    YES   NO

2-7 — ROLE ACCESS PERMITTED ?
  NO   YES

RETURN -1

2-3 — USER ACCESS PERMITTED ?
    YES   NO

TRY LISTED ROLES — 2-8
   OK   FAIL

2-9 — TRY GLOBAL ROLE
   OK   FAIL

2-4 — -T OPTION ? YES
  NO   RETURN 0   RETURN -1

2-5 — PERFORM ACTIONS

2-6 — EXECUTE TARGET COMMAND

RETURN RESULT

7

European Patent Office

EUROPEAN SEARCH REPORT

Application Number

EP 93 30 1288

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int. Cl.5)
X	EP-A-0 456 386 (INTERNATIONAL COMPUTERS LIMITED)	1-3	G06F1/00
Y	* figures 1,2,4 * * page 3, line 41 - line 55 * * page 4, line 9 - page 5, line 43 * ---	4-8	
Y	SOFTWARE PRACTICE & EXPERIENCE. vol. 20, no. 5, May 1990, CHICHESTER GB pages 485 - 497 M. BISHOP 'COLLABORATION USING ROLES' * page 486, line 40 - page 489, line 28 * * page 490, line 30 - page 491, line 29 * * page 493, line 7 - page 494, line 2 * ---	4-8	
A	US-A-5 077 795 (ROURKE ET AL.) * figures 8-13,16-25 * * column 7, line 24 - column 9, line 29 * -----	4-6	
			TECHNICAL FIELDS SEARCHED (Int. Cl.5)
			G06F

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	28 JUNE 1993	WEISS P.

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document